2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

WSCLoc: Weakly-Supervised Sparse-View Camera Relocalization via Radiance Field

Jialu Wang[1], Kaichen Zhou[1], Andrew Markham [1] and Niki Trigoni [1]

Abstract— **Despite the advancements in deep learning for camera relocalization tasks, obtaining ground truth pose labels required for the training process remains a costly endeavor. While current weakly supervised methods excel in lightweight label generation, their performance notably declines in scenarios with sparse views. In response to this challenge, we introduce WSCLoc, a system capable of being customized to various deep learning-based relocalization models to enhance their performance under weakly-supervised and sparse view conditions. This is realized with two stages. In the initial stage, WSCLoc employs a multilayer perceptron-based structure called WFT-NeRF to co-optimize image reconstruction quality and initial pose information. To ensure a stable learning process, we incorporate temporal information as input. Furthermore, instead of optimizing SE(3), we opt for $\mathfrak{sim}(3)$ optimization to explicitly enforce a scale constraint. In the second stage, we co-optimize the pre-trained WFT-NeRF and WFT-Pose. This optimization is enhanced by Time-Encoding based Random View Synthesis and supervised by inter-frame geometric constraints that consider pose, depth, and RGB information. We validate our approaches on two publicly available datasets, one outdoor and one indoor. Our experimental results demonstrate that our weakly-supervised relocalization solutions achieve superior pose estimation accuracy in sparse-view scenarios, comparable to state-of-the-art camera relocalization methods. We will make our code publicly available.**

I. Introduction

Deep learning-based camera relocalization, which is essential in fields like autonomous driving and augmented reality (AR), continues to be a prominent area of research. This technology employs neural networks to implicitly learn a map of a given scene, allowing for the estimation of free-trajectories for a moving camera based on captured images. While state-of-the-art methods achieve high accuracy, they heavily rely on manually annotating dense-view images, posing two main challenges: Firstly, additional sensors like RGB-D cameras or LiDAR are often required. Secondly, this process can be time-consuming and involve significant manual effort, sometimes necessitating a dedicated team for up to a year [1].

To achieve weakly-supervised camera relocalization without heavy handcrafted labels, recent advancements leverage Structure-from-Motion (SfM) techniques [2], [3] to automatically generate labeled images from RGB data alone, eliminating the need for additional sensors. However, this approach still relies on dense-view images, which are computationally demanding and impractical for consumer-grade devices due to resource constraints. The primary challenges stem from the lack of depth information, leading to scale drift, and image distortions like motion blur, significantly impacting performance in sparse-view scenarios.

In this study, we introduce Weakly-Supervised Sparse-View Camera Relocalization via Radiance Field (WSCLoc), a system designed to achieve weakly-supervised camera relocalization without heavy handcrafted labels and achieve state-of-the-art relocalization performance in sparse view scenarios. Our approach comprises two stages. In the initial stage, we utilize neural radiance techniques to generate pose labels under sparse-view conditions through our WFT-NeRF model. In the following stage, we introduce WFT-Pose, leveraging the previously generated pose labels for relocalization network training. Additionally, performance is enhanced through inter-frame geometric constraints from the pre-trained WFT-NeRF model. During the inference stage, this pose estimator enables rapid pose estimation for unseen images within the current scene. Our contributions can be summarized as follows:

- We propose a WFT-NeRF model that employs neural radiance techniques to generate pose labels from highly sparse views, particularly in scenarios with free-trajectories and large-scale settings, without the need for additional sensors.
- Leveraging the pose labels acquired in the initial stage, we propose WSCLoc, a system capable of being customized to various deep learning-based relocalization models to enhance their performance under weakly-supervised and sparse view conditions.
- Comprehensive experiments are conducted on a publicly available large-scale outdoor dataset and an indoor dataset to validate the effectiveness of our model.

II. Related Work

A. Weakly Supervised Camera Relocalization

Camera relocalization determines a camera's precise 6-DoF pose based on input images. Recent deep-learning methods, like CNNs [4], [2], [5], [6], [7], [8], offer efficient and accurate pose estimation. However, these methods rely on precise pose labels, requiring additional sensors (e.g., RGB-D cameras, LiDAR, GPS) and time-consuming data preprocessing (e.g., depth map alignment) [9]. In works such as [2], [3], weakly supervised relocalization has been successfully achieved using the Structure-from-Motion (SfM) [10] technique, but faces challenges in providing accurate pose labels due to issues like scale drift, image deformation in sparse-view scenarios.

[1]Jialu Wang, Kaichen Zhou, Niki Trigoni and Andrew Markham are with Department of Computer Science, University of Oxford, UK, jialu.wang@cs.ox.ac.uk, rui.zhou@kellogg.ox.ac.uk, niki.trigoni@cs.ox.ac.uk, andrew.markham@cs.ox.ac.uk

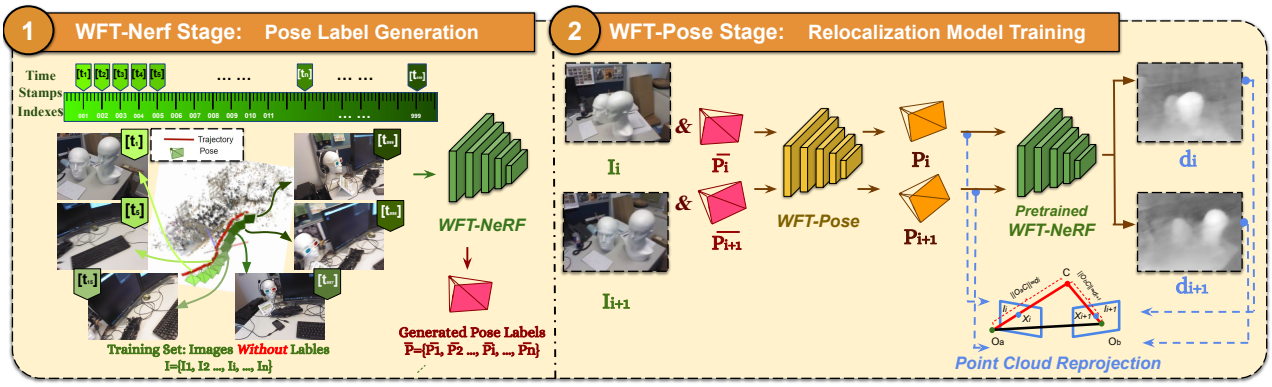

Fig. 1. **WSCLoc System Workflow.** In the WFT-NeRF Stage **(left)**, time encodings are generated for each image, and initial pose labels are obtained during the simultaneous training of the WFT-NeRF model. In the WFT-Pose Stage **(right)**, the training set is augmented using TE-based RVS (not shown in the figure). Consecutive frames are then fed into the target relocalization model in each iteration to calculate the pose loss and inter-frame geometric constraint loss. Finally, the relocalization model is trained by minimizing the overall loss.

B. Sparse-View Camera Relocalization

Recent advancements in camera pose estimation have yielded noteworthy contributions. For instance, RelPose [11] introduced a category-agnostic method for camera pose estimation, excelling particularly in rotation prediction but constrained by its ability to only predict rotations. Concurrently, SparsePose [12] pioneered camera pose regression followed by iterative refinement, while RelPose++ [13] decouples rotation estimation ambiguity from translation prediction through a novel coordinate system. However, these methods are confined to object-centric scenes. Moreover, [14] and PoseDiffusion [15], have shown promising results in camera pose estimation with very few annotated images. It's worth noting, though, that these methods still require additional sensors or dense-view structure-from-motion (SfM) reconstruction for image annotation. These research endeavors open new possibilities in camera pose estimation, yet underscore the need for further efforts to address data annotation requirements and applicability challenges in real-world scenarios.

C. NeRF with Pose Estimation

In recent research, there's a focus on eliminating the need for camera parameter preprocessing. Some methods like [16], [17], [18], [19], [20] rely on accurate camera poses from a SLAM tracking system or a pre-trained NeRF model. On the other hand, methods like [21], [22], [23], [24] go a step further by optimizing noisy camera poses during NeRF training. While these methods show promise for forward-facing datasets, they are often limited to handling forward-facing scenes or 360° object-centric unbounded scenes.

III. Methods

The unsupervised camera relocalization is realized by WSCLoc with two steps. During the first step, WFT-NeRF is trained to generate initial pose information. During the second step, WFT-Pose is co-optimized with the WFT-NeRF to realize accurate pose estimation for unseen images.

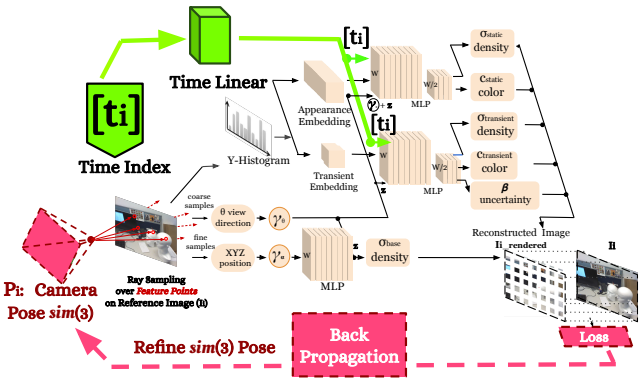

Fig. 2. **Structure of WFT-NeRF.** During video capture, reference images are encoded with temporal information (t_i) using discrete time indices to minimize motion-related blurring. Grayscale levels in YUV are encoded for consistent exposure and appearance. Our NeRF training involves three sets of MLPs: 1. The base network estimates volume density and hidden state after coarse ray sampling. 2. Middle MLPs perform fine-ray sampling for appearance, estimating density and color. 3. Top MLPs handle fine-ray sampling for transient properties, estimating density, color, and uncertainty to filter transient objects. Losses between the rendered and reference images optimize pose during backpropagation, simultaneously optimizing NeRF and $\mathfrak{sim}(3)$ poses. Only the base network is used for testing.

A. Weakly-supervised Free-Trajectory (WFT) NeRF

While methods like [22], [21] can refine noisy camera poses during NeRF training, they are inadequate for large-scale or free-trajectory scenes. The NeRF model from [6], [25] can handle such scenarios but lacks direct pose optimization capabilities. This is primarily because RGB images from monocular cameras can introduce scale drift, and artifacts may occur during NeRF model training due to distortions in images, such as rolling shutter effects, object deformations, or motion blur [26]. To tackle these challenges, we introduced two key enhancements: 1. Explicit Scale Constraint to mitigate scale drifting. 2. Explicit Time Encoding to handle image distortions like boundary blurriness.

1) Explicit Scale Constraint to Address Scale Drifting: To mitigate scale drift, we introduce an additional scale factor s using similarity transformations instead of Euclidean

transformations ($SE(3)$) [27]. Given an initial SE(3) pose, we represent the transformation as a 4×4 matrix (see Eq. (1)), where $\mathbf{R}$, $\mathbf{t}$, and s denote the rotation matrix, translation matrix, and scale factor, respectively. To maintain the estimated pose within the pose manifold during gradient-based optimization, we parameterize it using exponential coordinates.

$$\mathfrak{sim}(3) = \begin{bmatrix} s\mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \in \mathbb{R}^{4*4}, \tag{1}$$

To maintain the estimated pose within the pose manifold during gradient-based optimization, we parameterize it using exponential coordinates. Thus, we represent the transformation as a 7-dimensional sim(3) vector, as shown in Eq. (2). Their exponential mapping relationships are described in Eq. (3, 4), where the axis-angle representation is defined as $\phi = \theta\mathbf{a}$, $\mathbf{a}$ is a unit direction vector, and $\theta = \arccos \frac{\mathrm{tr}(\mathbf{R})-1}{2}$ is the magnitude [27].

$$\mathfrak{sim}(3) = \left\{ \mathbf{P} \mid \mathbf{P} = \begin{bmatrix} \boldsymbol{\rho} \\ \boldsymbol{\phi} \\ \sigma \end{bmatrix} \in \mathbb{R}^7 \right\}, \tag{2}$$

$$s = e^{\boldsymbol{\rho}}, \mathbf{R} = \exp(\boldsymbol{\phi}^\wedge), \mathbf{t} = J_s\boldsymbol{\rho}, \tag{3}$$

$$\begin{aligned} &\text{,where } J_s = \frac{e^\sigma - 1}{\sigma} + \frac{\sigma e^\sigma \sin\boldsymbol{\theta} + (1 - e^\sigma \cos\boldsymbol{\theta})\boldsymbol{\theta}}{\sigma^2 + \boldsymbol{\theta}^2} \\ &+ (\frac{e^\sigma - 1}{\sigma} - \frac{(e^\sigma \cos\boldsymbol{\theta} - 1)\sigma + (e^\sigma \sin\boldsymbol{\theta})\boldsymbol{\theta}}{\sigma^2 + \boldsymbol{\theta}^2})\mathbf{a}^\wedge\mathbf{a}^\wedge, \end{aligned} \tag{4}$$

In addition, considering that almost all NeRF methods utilize R and t to jointly render volume density but additionally employ R to render color, we opted to independently optimize the translation and rotation components at different training rates, rather than equally optimizing the entire 7-dimensional $\mathfrak{sim}(3)$ vector.

$$\bar{\boldsymbol{\rho}}\bar{\boldsymbol{\phi}}\bar{\sigma} = \underset{\bar{\boldsymbol{\rho}}\bar{\boldsymbol{\phi}}\bar{\sigma} \in \mathbb{R}^7}{\mathrm{argmin}} L(\bar{\mathbf{P}} \mid I, \boldsymbol{\Theta}), \tag{5}$$

In the end, the problem of optimizing poses can be expressed using (Eq. (4)). Let Θ represent the parameters of WFT-NeRF, $\bar{\mathbf{P}}_i = e^{\sigma_i} \exp(\boldsymbol{\phi}_i^\wedge)\bar{\mathbf{P}}_0$ denote the estimated camera pose at the current optimization step i, I represent the observed image, and L be the loss used for training WFT-NeRF. Our goal is to determine the optimal pose starting from an initial estimate $\bar{\mathbf{P}}_0$ (in the sparse-view scenario, we obtain initial noisy pose estimates $\bar{\mathbf{P}}_0$ from SfM techniques). To achieve this, we employ gradient descent with the Adam optimizer to independently optimize the three components ($\boldsymbol{\rho}$, $\boldsymbol{\phi}$, and σ) of the $\mathfrak{sim}(3)$ pose using the photometric loss function L of NeRF.

2) Explicit Time Encoding to Addresses Boundary Blurriness: Large-scale or free-trajectories scenarios can easily introduce camera rolling shutter effects and motion blur, which may lead to artifacts during the training of the NeRF model (Fig. 3), as demonstrated in [26], ultimately resulting in incorrect pose estimations.

To mitigate this issue, we explicitly encode discrete time indices for the input images (as shown in Fig. 2). This introduces additional timestamp constraints to the pixel positions for each frame, effectively enforcing the camera's impact on objects to be smoother and reducing the boundary blurring caused by the camera's motion. As depicted in Fig. 3, this enables us to rectify deformed ground truth images, rendering sharper object boundaries, thereby enhancing robustness to motion blur.

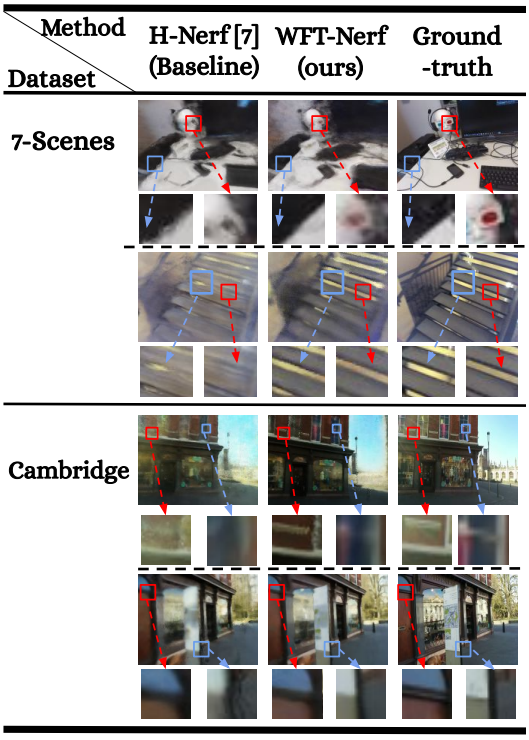

Fig. 3. **Qualitative Comparison.** Large-scale or free-trajectories can introduce camera rolling shutter effects and motion blur, causing artifacts during NeRF model training and resulting in incorrect pose estimations. We mitigate this by explicitly encoding time indices for input images, rectifying deformed ground truth images, and enhancing robustness to motion blur with sharper object boundaries.

3) WFT-NeRF Structure Design: The training process of WFT-NeRF is shown in Fig. 2. To handle poses for large-scale or free-trajectories scenes, our WFT-NeRF structure is built upon a state-of-the-art NeRF model, showcased in [6]. In addition to incorporating our proposed techniques mentioned in III-A.2 III-A.1, we made two additional adjustments to further enhance its effectiveness: 1. To stabilize camera trajectory optimization in large scenes and ensure accurate registration, we replaced the original full positional encoding with a softer variant from [3]. This variant begins with a smooth signal and gradually shifts focus to learn a high-fidelity scene representation. We strongly recommend readers to refer to [6], [3] for more details. 2. To reduce computational costs, [6], [19], [22] employ partial ray sampling for faster training while maintaining accuracy. Our approach stands out by using SIFT feature extraction [28] to identify and match feature points between adjacent frames, enhancing rendering efficiency and performance in feature-rich areas for weakly supervised relocalization network training.

B. Weakly-supervised Free-Trajectory Pose (WFT-Pose)

In this section, we introduce WFT-Pose, a system capable of being customized to various deep learning-based relocalization models to enhance their performance under weakly-supervised and sparse view conditions. When applied to a target relocalization model, WFT-Pose optimizes it alongside a pre-trained WFT-NeRF model, integrating additional inter-frame geometric constraints and Time-Encoding Based Random View Synthesis (TE-based RVS) data augmentation to ensure accurate pose estimation for unseen images.

1) Inter-Frame Geometric Constraints: To compensate for the lack of scale information and mitigate scale drift issues in the input images from monocular cameras, we used depth maps generated by the WFT-NeRF to compute a point cloud loss based on KL divergence between adjacent frames. During this process, WFT-NeRF is pre-trained, and its weights are frozen.

During the training phase, for each pair of adjacent images (denoted as $\{I_i,\ I_{i+1}\}$, we perform feature matching to obtain a set of normalized coordinates of feature points $\{X_i,\ X_{i+1}\}$ on the image plane by employing SIFT feature extraction [28] and nearest-neighbor search based on k-dimensional trees (k-d trees) [29].

Then we back-project the depth map $\{d_i,\ d_{i+1}\}$ rendered by the pre-trained WFT-NeRF using the estimate poses $\{P_i,\ P_{i+1}\}$ obtained by the target relocalization model, to point clouds $\{C_i,\ C_{i+1}\}$ and optimise the relative pose between consecutive point clouds by minimising the point cloud loss L_{pc} (see Eq. (6, 7)), where $P_{i+1,i} = P_{i+1}^{-1} \cdot P_i$ represents the relative pose that transforms point cloud C_i to C_{i+1}.

$$L_{pc} = \sum_{(i,\ i+1)} (C_{i+1},\ P_{(i+1,\ i)}\ C_i) \tag{6}$$

$$= \sum_{(i,\ i+1)} (d_{i+1}X_{i+1}P_{i+1},\ P_{(i+1,\ i)}\ d_iX_iP_i), \tag{7}$$

Since d_i and d_{i+1} inherently contain noise, and the feature points are quite sparse, we additionally introduce a KL divergence loss to further reduce the discrepancy between the distributions of the two frames. The final inter-frame geometric constraint loss (abbreviated as IF Loss) is shown in (Eq. (8)), where $\mathcal{P}_{i+1}$ and $\mathcal{P}_i$ are the probability distributions of C_{i+1} and $P_{(i+1,\ i)}\ C_i$, respectively.

$$L_{IF} = L_{pc} + \sum_{(i,\ i+1)} (KL[\mathcal{P}_{i+1}, \mathcal{P}_i]), \tag{8}$$

2) Time-Encoding Based Random View Synthesis (TE-Based RVS): To enhance generalization, we adopt a strategy akin to [30], [6] for generating additional training data by perturbing poses with NeRF. In contrast to previous methods, we utilize pretrained WFT-NeRF to introduce additional time encoding to these perturbed poses, thereby improving image sharpness and augmenting the effectiveness of data augmentation. TE-based RVS poses are generated around the training pose with a random translation and rotation noise of 0.2m and 10° respectively. For spaese-view scenarios with 20% and 10% of the train set images, we generate synthetic images at scales of 5, and 10 times the original number of images to ensure an equal scale of training images across all conditions.

C. Weakly-supervised camera relocalization (WSCLoc)

Expanding on our WFT-NeRF and WFT-Pose frameworks, we introduce WSCLoc, a versatile system designed to be capable of being customized to various deep learning-based relocalization models to enhance their performance under weakly-supervised and sparse view conditions. When integrated into a target relocalization model, WSCLoc first generates pose labels by training the WFT-NeRF model. Subsequently, it co-optimizes the target relocalization model with pre-trained WFT-NeRF to achieve accurate pose estimation for unseen images. The detailed workflow of WSCLoc is outlined below:

Training Phase of the WSCLoc (Fig. 1) is as follows:

1) **Image preprocessing** Given a series of RGB images, generate time encodings $t_i, t_j, ...$ for each image, and perform feature matching between adjacent frames.
2) **WFT-NeRF Stage: Pose Label Generation** Generating initial pose labels $\{\bar{P}_i, \bar{P_{i+1}}, ...\}$ during the simultaneous training of the WFT-NeRF model.
3) **WFT-Pose Stage: Relocalization Model Training** First, augment the training set using TE-based RVS. Then, in each iteration, feed two consecutive frames (I_i, I_{i+1}) into the target relocalization model to calculate the pose loss and our inter-frame geometric constraint loss (Eq. (8)). The relocalization model is trained by minimizing the overall loss.

During the testing phase, we retain only the relocalization network for pose estimation.

IV. EXPERIMENTS

A. Relocalization Models and Evaluation Metrics

To showcase the versatility of our system, we apply our WSCLoc to enhance two models for weakly supervised relocalization: DFNet [6], a state-of-the-art NeRF-based model, and PoseNet [2], the most traditional end-to-end relocalization model. In this study, we adhere to the same evaluation strategies and datasets as the original papers for all models to ensure a fair comparison. Specifically, we evaluate the median translation error (m) and median rotation error (degree) for all models.

B. Datasets

We assess the effectiveness of our method using two widely-used datasets. The **7-Scenes Dataset** [31] encompasses seven indoor scenes featuring RGB images, depth maps, and ground truth camera poses obtained from KinectFusion. The **Cambridge Landmarks Dataset** [2] comprises six large-scale outdoor scenes with RGB images, visual reconstructions of each scene, and 6-DoF ground truth camera poses reconstructed using SfM. Notably, due to inaccuracies in the 3D reconstruction of the STREET scene in Cambridge, consistent with prior studies [32], [6], our experiments are conducted solely on the remaining five scenes.

C. Implementation Details

We trained all models under the dense-view condition (trained with 100% of the train set images) and two sparse-view conditions (trained with 20% and 10% of the train set images), and then tested them on the complete (100% images) test set. In the sparse-view experiments, we obtained the training set images by uniformly sampling every $5th$ and $10th$ image to achieve 20% and 10% of the training set, respectively, and then generated corresponding pose labels using SfM. It is worth noting that we chose the 20% and 10% image settings for experimentation because SfM fails to work properly in many scenes of these datasets when fewer than 10% of the images are used. We utilized COLMAP [33], one of the most popular Structure-from-Motion (SfM) tools, to generate train set pose labels for all scenes under sparse-view conditions. To ensure fairness in comparison, we maintained consistent hyperparameters across all models, following the settings used in the original papers. For training the WFT-NeRF models, we set the learning rate to 0.001. Training was conducted until convergence on a single Nvidia RTX-3090 GPU.

D. WSCLoc Performance Evaluation

As WSCLoc is capable of being customized to various deep learning-based relocalization models to enhance their performance under weakly-supervised and sparse view conditions, we leverage our WSCLoc approach to enhance the DFNet [6] model, a state-of-the-art NeRF-based model, for weakly supervised relocalization, and compare its performance with that of the origional DFNet. Additionally, we conducted supplementary experiments on PoseNet [2], a conventional end-to-end relocalization model, to demonstrate the versatility of our approach.

1) Evaluation on DFNet [6] Model: In this section, we leverage our WSCLoc approach to enhance the DFNet model for weakly supervised relocalization.In the WFT-NeRF training stage, in sparse-view scenarios, we simultaneously generate pose labels while training the WFT-NeRF. Subsequently, in the WFT-Pose stage, we integrate our NVS and inter-frame geometric constraints into the vanilla PoseNet model. We then benchmark against the origional DFNet [6] and conduct experiments on both the large-scale outdoor dataset (Cambridge Landmark [2]) and the indoor dataset (7scenes [31]) to evaluate their performances.

Evaluation on 7-Scenes Dataset The experimental results on the 7-Scenes indoor dataset with complex trajectories are shown in Table I. In sparse-view scenarios, as mentioned in Section I, the pose estimation accuracy from SfM tends to be noisier, resulting in lower relocalization accuracy for baseline methods (e.g., DFNet) compared to dense-view conditions. In contrast, our WSCLoc achieves relocalization accuracy comparable to dense-view conditions, even outperforming the benchmarks in some scenarios with only 10% of the images (e.g., fire, redkitchen). In dense-view scenarios, our method achieves comparable results to the baseline. However, in certain cases, the baseline outperforms ours, possibly due to errors introduced by incorrect feature matching during FT Loss computation.

Evaluation on Cambridge Dataset We then evaluated our method on a more challenging large-scale outdoor dataset. The experimental results are summarized in Table II. In the sparse-view scenario, all relocalization accuracies of the baseline noticeably degrade. In contrast, our WSCLoc method outperforms the baseline in all sparse-view scenarios, particularly achieving a 51% and 33% improvement over the averaged median translation and rotation errors in the 10% images scenario compared to the baseline performance. In dense-view scenarios, our method achieves comparable results to the baseline, but in some cases, the baseline outperforms ours due to potential errors in feature matching during FT Loss computation.

2) Evaluation on PoseNet [2] Model: To demonstrate the versatility of our WSCLoc system, we applied it to the classic end-to-end relocalization model, PoseNet [2], using the methodology outlined in Section IV-D.1. Following this, we conducted comparative experiments on the large-scale Cambridge dataset against the baseline PoseNet model. Table III illustrates a notable decline in pose accuracy for the vanilla PoseNet with increasing view sparsity. In contrast, our WSCLoc method achieves results comparable to those obtained in dense-view scenarios. This indicates the effectiveness of WSCLoc in addressing the challenges posed by sparse views in camera relocalization tasks.

E. WFT-NeRF Performance Evaluation

1) Quantitative Results for Pose Label Generation: This section quantitatively demonstrates the performance of WFT-NeRF in generating pose labels in sparse-view scenarios. We obtained 20% and 10% of the train set images by uniformly sampling every $5th$ and $10th$ image, respectively, and then used SfM to generate pose labels in these two sparse-view scenarios. Subsequently, we compared the median translation and rotation errors between the pose labels generated by WFT-NeRF and the precise pose labels generated by SfM in the dense-view (100% train set images) scenario.

The experimental results are shown in Table IV. Compared to the precise poses obtained in the dense-view scenario, the pose error generated by SfM in both sparse-view scenarios with 20% and 10% of the images are significantly larger, and the errors increase as the sparsity increases. In contrast, the median translation and rotation errors of poses generated by WFT-NeRF are much closer to the ground truth poses botained in the dense-view scenario, demonstrating the ability of WFT-NeRF to effectively produce pose labels close to dense-view level even in highly sparse-view scenarios.

2) QualitativeResults for Image Rendering: This section qualitatively demonstrates the performance of WFT-NeRF in rendering images in large-scale and complex trajectories scenes. We employed DFNet's Histogram-assisted NeRF [6], a state-of-the-art NeRF variant capable of operating in large-scale or free-trajectory scenes, as our baseline. We conducted experiments on both the large-scale Cambridge dataset and the 7-Scenes dataset with complex trajectories.

TABLE I

PERFORMANCE EVALUATION OF WSCLOC ON THE DFNET MODEL [6] USING THE 7-SCENES DATASET [31]. WE FOLLOW THE ORIGINAL PAPERS' EVALUATION PROTOCOLS TO ENSURE FAIRNESS. SPECIFICALLY, WE MEASURE MEDIAN TRANSLATION AND ROTATION ERRORS IN $m/^\circ$.

%Imgs	Method	Chess	Fire	Heads	Office	Pumpkin	Kitdhen	Stairs
100%	DFNet[7]	0.05/1.88	0.17/6.45	**0.06/3.63**	**0.08/2.48**	0.10/2.78	0.22/5.45	**0.16**/3.29
	WSCLoc	0.05/**1.83**	**0.13/4.80**	0.07/3.88	0.13/2.99	0.10/**2.33**	**0.14/3.83**	0.18/**3.20**
20%	DFNet[7]	0.18/6.30	0.41/10.12	0.21/14.04	0.82/9.43	0.40/8.51	0.51/12.82	0.79/12.20
	WSCLoc	**0.05/1.84**	**0.13/4.97**	**0.07/3.90**	**0.12/3.24**	**0.10/2.46**	**0.14/3.95**	**0.18/3.68**
10%	DFNet[7]	0.19/9.23	0.52/10.88	0.21/16.17	0.90/10.21	0.44/9.01	0.60/13.08	0.79/14.11
	WSCLoc	**0.05/1.86**	**0.17/5.33**	**0.13/7.19**	**0.14/3.78**	**0.11/2.90**	**0.14/3.76**	**0.22/4.71**

TABLE II

PERFORMANCE EVALUATION OF WSCLOC ON THE DFNET MODEL [6] USING THE CAMBRIDGE DATASET [2]. WE MEASURE MEDIAN TRANSLATION AND ROTATION ERRORS IN $m/^\circ$.

%Imgs	Method	Kings	Hospital	Shop	Church
100%	DFNet[7]	0.73/2.37	**2.00**/2.98	**0.67/2.21**	**1.37**/4.03
	WSCLoc	**0.71/2.07**	**2.00/2.79**	1.02/2.92	1.52/**3.98**
20%	DFNet[7]	1.93/8.02	4.12/8.30	2.11/10.05	5.30/12.12
	WSCLoc	**1.31/2.59**	**2.94/3.61**	**1.38/5.85**	**3.16/6.73**
10%	DFNet[7]	2.39/8.77	4.30/9.78	3.47/12.20	7.02/14.77
	WSCLoc	**1.69/2.74**	**3.47/3.85**	**2.48/8.64**	**3.78/7.73**

TABLE III

PERFORMANCE EVALUATION OF WSCLOC ON THE POSENET MODEL [2] USING THE CAMBRIDGE DATASET [2]. WE MEASURE MEDIAN TRANSLATION AND ROTATION ERRORS IN $m/^\circ$.

% Imgs	Method	Avg. Pose Error
100 %	PoseNet [2]	2.04/6.23
	WSCLoc (PN)	**1.99/5.18**
20 %	PoseNet [2]	4.85/18.62
	WSCLoc (PN)	**2.47/6.33**
10 %	PoseNet [2]	8.02/20.38
	WSCLoc (PN)	**2.82/7.41**

TABLE IV

THE ERROR BETWEEN GENERATED POSE LABELS UNDER SPARSE-VIEWS AND THOSE GENERATED UNDER DENSE-VIEW (100%) BY SFM. WE MEASURE MEDIAN TRANSLATION AND ROTATION ERRORS IN $m/^\circ$.

Data Set	% Imgs	SfM	WFT-NeRF
7 Scenes	20 %	1.52/7.11	**0.05/0.31**
	10 %	1.64/7.83	**0.07/0.41**
Cambridge	20 %	1.16/9.12	**0.02/0.53**
	10 %	1.70/13.36	**0.06/0.72**

Figure 3 qualitatively demonstrates the rendering performance of our WFT-NeRF and the baseline in extremely sparse-view scenarios on these challenging datasets. The baseline exhibits blurry boundaries in rendered images when significant changes occur in camera poses. In contrast, our WFT-NeRF produces clearer boundaries, which is beneficial for subsequent tasks such as WSCLoc implementation of TE-based random view synthesis and depth-map rendering. The PSNR results of NeRF can be found in the ablation study.

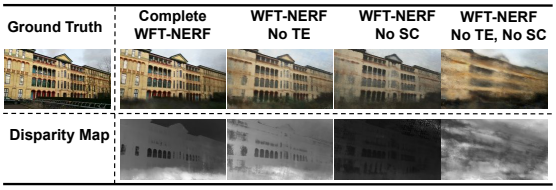

Fig. 4. **Qualitative Evaluation of WFT-NeRF Performance** (Hospital scene in the large-scale Cambridge dataset under 10% sparse-view conditions). Here, we demonstrate the impact of our full WFT-NeRF model benefiting from explicit scale constraint (SC) and Time Encoding (TE). Removing TE alone results in less clear image boundaries. Removing SC only leads to blurry images due to noisy pose labels generated by SfM in sparse-view scenarios. Removing both TE and SC severely degrades the quality of rendered images.

F. Ablation Study

1) Effectiveness of Explicit Scale Constraint and Time Encoding of WFT-NeRF: We evaluated the effectiveness of Explicit Scale Constraint (SC) and Time Encoding (TE) in WFT-NeRF under highly sparse-view conditions on the Cambridge Dataset. We trained the model using 10% of the train set images and tested the rendering performance of WFT-NeRF on the complete test set. As shown in Fig. 4, the results of qualitative experiments demonstrate that the Explicit Time Encoding effectively improves the clarity of object boundaries in rendering. From the quantitative results in Table V, we found that in sparse-view scenarios, the Explicit Scale Constraint provides greater assistance in improving PSNR by generating more accurate poses.

2) Effectiveness of TE-based RVS and inter-frame geometric constraints of WFT-Pose: We also evaluated the effectiveness of TE-based RVS and inter-frame geometric constraints (IF Loss) in WFT-Pose under the same experimental settings as IV-F.1. As shown in Table V, we found that in highly sparse-view scenarios, the inter-frame geometric constraints provides less assistance in reducing pose errors compared to TE-based RVS. We attribute this to the sparsity of feature points matched between adjacent frames in highly sparse-view scenarios, which weakens the geometric constraints for relocalization. In future work, we will consider using deep learning-based feature matching methods to further optimize its performance.

TABLE V

ABLATION STUDY 10% IMGS ON CAMBRIDGE DATASET. WE MEASURE MEDIAN TRANSLATION AND ROTATION ERRORS IN $m/^\circ$.

WFT-NeRF			WFT-Pose		
SC	TE	PSNR	TE-based RVS	IF Loss	Avg. Pose Error
		16.20			3.44/6.20
✓		19.87	✓		2.88/5.90
	✓	16.80		✓	2.92/6.03
✓	✓	**20.95**	✓	✓	**2.85/5.74**

V. CONCLUSIONS

In summary, traditional deep learning-based camera relocalization models heavily rely on manually annotating dense-view images. While existing weakly-supervised methods excel in lightweight label generation, their performance significantly degrades in sparse-view scenarios. To address these challenges, we introduce WSCLoc, a system capable of being customized to various deep learning-based relocalization models to enhance their performance under weakly-supervised and sparse view conditions. WSCLoc first generates pose labels by training our WFT-NeRF, then co-optimizes the target relocalization model with the pre-trained WFT-NeRF to achieve accurate pose estimation for unseen images. Our innovations include explicit scale constraint and time encoding of our WFT-NeRF, as well as Time-Encoding Based Random View Synthesis and Inter-Frame Geometric Constraints of our WFT-Pose. Experimental results on diverse datasets demonstrate that our weakly-supervised solutions achieve state-of-the-art accuracy performance in sparse-view scenarios.

REFERENCES

[1] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The oxford robotcar dataset," *The International Journal of Robotics Research*, vol. 36, no. 1, pp. 3–15, 2017.

[2] A. Kendall, M. Grimes, and R. Cipolla, "Posenet: A Convolutional Network for Real-time 6-DOF Camera Relocalization," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2938–2946.

[3] S. Chen, Z. Wang, and V. Prisacariu, "Direct-posenet: absolute pose regression with photometric consistency," in *2021 International Conference on 3D Vision (3DV)*. IEEE, 2021, pp. 1175–1185.

[4] B. Wang, C. Chen, C. X. Lu, P. Zhao, N. Trigoni, and A. Markham, "Atloc: Attention guided camera localization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 06, 2020, pp. 10 393–10 401.

[5] S. Brahmbhatt, J. Gu, K. Kim, J. Hays, and J. Kautz, "Geometry-aware learning of maps for camera localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2616–2625.

[6] S. Chen, X. Li, Z. Wang, and V. A. Prisacariu, "Dfnet: Enhance absolute pose regression with direct feature matching," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part X*. Springer, 2022, pp. 1–17.

[7] J. Liu, Q. Nie, Y. Liu, and C. Wang, "Nerf-loc: Visual localization with conditional neural radiance field," *arXiv preprint arXiv:2304.07979*, 2023.

[8] K. Zhou, C. Chen, B. Wang, M. R. U. Saputra, N. Trigoni, and A. Markham, "Vmloc: Variational fusion for learning-based multimodal camera localization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 7, 2021, pp. 6165–6173.

[9] K. Zhou, L. Hong, C. Chen, H. Xu, C. Ye, Q. Hu, and Z. Li, "Devnet: Self-supervised monocular depth learning via density volume construction," in *European Conference on Computer Vision*. Springer, 2022, pp. 125–142.

[10] S. Ullman, "The interpretation of structure from motion," *Proceedings of the Royal Society of London. Series B. Biological Sciences*, vol. 203, no. 1153, pp. 405–426, 1979.

[11] J. Y. Zhang, D. Ramanan, and S. Tulsiani, "Relpose: Predicting probabilistic relative rotation for single objects in the wild," *arXiv e-prints*, pp. arXiv–2208, 2022.

[12] S. Sinha, J. Y. Zhang, A. Tagliasacchi, I. Gilitschenski, and D. B. Lindell, "Sparsepose: Sparse-view camera pose regression and refinement."

[13] A. Lin, J. Y. Zhang, D. Ramanan, and S. Tulsiani, "Relpose++: Recovering 6d poses from sparse-view observations," *arXiv preprint arXiv:2305.04926*, 2023.

[14] S. Dong, S. Wang, Y. Zhuang, J. Kannala, M. Pollefeys, and B. Chen, "Visual localization via few-shot scene region classification," *arXiv preprint arXiv:2208.06933*, 2022.

[15] J. Wang, C. Rupprecht, and D. Novotny, "Posediffusion: Solving pose estimation via diffusion-aided bundle adjustment," *arXiv e-prints*, pp. arXiv–2306, 2023.

[16] A. Rosinol, J. J. Leonard, and L. Carlone, "Nerf-slam: Real-time dense monocular slam with neural radiance fields," *arXiv preprint arXiv:2210.13641*, 2022.

[17] E. Sucar, S. Liu, J. Ortiz, and A. J. Davison, "imap: Implicit mapping and positioning in real-time," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6229–6238.

[18] Z. Zhu, S. Peng, V. Larsson, W. Xu, H. Bao, Z. Cui, M. R. Oswald, and M. Pollefeys, "Nice-slam: Neural implicit scalable encoding for slam," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 786–12 796.

[19] L. Yen-Chen, P. Florence, J. T. Barron, A. Rodriguez, P. Isola, and T.-Y. Lin, "inerf: Inverting neural radiance fields for pose estimation," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 1323–1330.

[20] K. Zhou, J.-X. Zhong, S. Shin, K. Lu, Y. Yang, A. Markham, and N. Trigoni, "Dynpoint: Dynamic neural point for view synthesis," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[21] Z. Wang, S. Wu, W. Xie, M. Chen, and V. A. Prisacariu, "Nerf–: Neural radiance fields without known camera parameters," *arXiv preprint arXiv:2102.07064*, 2021.

[22] C.-H. Lin, W.-C. Ma, A. Torralba, and S. Lucey, "Barf: Bundle-adjusting neural radiance fields," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5741–5751.

[23] Y. Jeong, S. Ahn, C. Choy, A. Anandkumar, M. Cho, and J. Park, "Self-calibrating neural radiance fields," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5846–5854.

[24] W. Bian, Z. Wang, K. Li, J.-W. Bian, and V. A. Prisacariu, "Nope-nerf: Optimising neural radiance field with no pose prior," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4160–4169.

[25] P. Wang, Y. Liu, Z. Chen, L. Liu, Z. Liu, T. Komura, C. Theobalt, and W. Wang, "F2-nerf: Fast neural radiance field training with free camera trajectories," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4150–4159.

[26] V. Prisacariu, S. Chen, and Z. Wang, "Direct-posenet: Absolute pose regression with photometric consistency," 2022.

[27] H. Strasdat, J. Montiel, and A. J. Davison, "Scale drift-aware large scale monocular slam," *Robotics: science and Systems VI*, vol. 2, no. 3, p. 7, 2010.

[28] D. G. Lowe, "Object recognition from local scale-invariant features."

[29] J. H. Friedman, J. L. Bentley, and R. A. Finkel, *An algorithm for finding best matches in logarithmic time*. Department of Computer Science, Stanford University, 1975.

[30] A. Moreau, N. Piasco, D. Tsishkou, B. Stanciulescu, and A. de La Fortelle, "Lens: Localization enhanced by nerf synthesis," in *Conference on Robot Learning*. PMLR, 2022, pp. 1347–1356.

[31] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon, "Scene coordinate regression forests for camera relocalization in rgb-d images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 2930–2937.

[32] E. Brachmann and C. Rother, "Visual camera re-localization from rgb and rgb-d images using dsac," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[33] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited."